REMARKS/ARGUMENTS

In the Advisory Action, the Examiner indicated that the period for reply expires 3 months from the mailing date of the Final Office Action of April 15, 2006. However, the Response to the Advisory Action was submitted on Monday, April 17th, which is two months from the mailing date of April 15th of the Final Office Action because the due date of April 15th fell on a weekend and the next business day was Monday, April 17th. The Examiner acknowledged this mistake in a phone conversation and agreed that the period for reply should expire on the mailing date of the Advisory Action (June 22, 2006). The Examiner said he submitted a new Advisory Action indicating the proper period for reply and having a new expiration date. However, as of this date, Applicants have not received this new Advisory Action nor found it listed in PAIR. For this reason, Applicants calculate the period for reply as expiring on the mailing date of the June 22nd Advisory Action.

1. Amended Claims 1 and 15 Comply with the Enablement and Definiteness Requirements

The Examiner rejected claims 1 and 15 as not satisfying the enablement requirement of 35 U.S.C. §112, par. 1 on the grounds the language that "the mapping is used to map and convert files from the host file system to the local file system to support remote editing" is not supported by the Specification. (Final Office Action, pg. 2) Applicants traverse with respect to the amended claims.

Although Applicants disagree with the basis for the Examiner's enablement rejection as set forth in the Response to Final Office Action dated February 15, 2006, Applicants amended claims 1 and 15 to remove the wherein phrase that the Examiner found unsupported by the Specification and amended the mapping data structure element to now recite:

> a mapping data structure comprising at least one tag representing a mapping between files in the local file system and files in the host file system and a transfer type that defines a data format for transferring data between the host system and the local system for to support remote editing of files in the host file system from the local file system

Applicants amended claims 1 and 15, as well as claim 2, to clarify that the at least one tag represents a mapping between a file in the local file system and a file in the host file system.

Applicants submit that this amendment is disclosed in the Specification on at least pg. 12, lines 25-29, which discloses "a mapping between a file within the host directory path on the host file system and a file on the local file system."

Applicants further added the requirement of "a transfer type that defines a data format for transferring data between the host system and the local system for this mapping." The Specification discloses "a transfer type that defines how data will be transferred between host and workstation for this mapping." (Specification, pg. 16, lines 10-17). The Specification specifically discloses different data format types that may be defined for the "transfer type", such as "text" and "binary", which are data format types. (Specification, pg. 15, line 28 to pg. 16, line 1; pg. 16, lines 11-17; pg. 18, lines 24-27; pg. 23, lines 15-28). Moreover, Applicants note that this added limitation is found in claim 3, which prior to the amendment recited a transfer type limitation.

Applicants further submit that the Specification discloses the claim requirement that the mapping is used to " to support remote editing of files in the host file system from the local file system". See, Specification, pg. 12, lines 17-20 ("The present invention provides for such access and use of host files on the workstation **12** by providing a data structure for configuring a connection such that files and directories may be mapped and converted from the host **18** to the workstation **12** to support a scenario such as the Remote Edit/Compile/Debug"). According to the Specification, "Remote Edit/Compile/Debug provides a workstation environment for performing the edit, compile, and debug tasks associated with host application development." (Specification, pg. 3, lines 18-19, pg. 12, lines 10-14)

Accordingly, Applicants submit that the added claim requirements are supported by the disclosure of the Specification.

2. Claims 1-21 are Patentable Over the Cited Art

The Examiner rejected claims 1-21 as obvious (35 U.S.C. §103) over Stedman (U.S. Patent No. 6,081,837), Imai (U.S. Patent No. 6,148,334), and Harvey (U.S. Patent No. 6,519,568). Applicants traverse.

Claims 1 and 15 concern providing information describing a file system connection between a local file system located on a local system and a host file system located on a host

system, said method comprising: encoding the information in a metalanguage format comprising one or more tags, each tag having an identifier and a set of one or more attributes, wherein the encoded information comprises a file system connection descriptor; said file system connection descriptor comprising: a local system data structure comprising at least one tag representing the local file system; a host system data structure comprising at least one tag representing the host file system; and a mapping data structure comprising at least one tag representing a mapping between files in the local file system and files in the host file system and a transfer type that defines a data format for transferring data between the host system and the local system to support remote editing of files in the host file system from the local file system; and parsing the file system connection descriptor according to the metalanguage tags.

The Examiner cited col. 19, lines 53-67 and col. 20, lines 1-25 of Stedman as teaching the claim requirement of encoding the information in a metalanguage format comprising one or more tags, each tag having an identifier and a set of one or more attributes. (Final Office Action, pg. 3) Applicants traverse.

Stedman concerns a way to transfer information between a host and a client by having a host extension a create set of instructions that is transferred to the client computer. The client computer utilizes the set of instructions to create web pages displayed at the client computer. (Stedman, col. 2, lines 47-60)

The cited col. 19 mentions that a host extension creates an HTML document and inserts URLs within the document and tags that identify where the session ID is placed. Stedman defines the host extension as receiving information from a host computer and creating a set of instructions that is transferred via the server application framework to the client computer. The client computer utilizes these instructions to create Web pages. (Stedman, col. 2, lines 47-55) The cited col. 19 further mentions that the server application framework replaces session ID tags with the appropriate session ID.

The cited col. 19 discusses how a host extension creates an HTML document with tags for the session ID to send to the client system. Nowhere does this cited col. 19 of Stedman anywhere teach or suggest that the host extension encode information on a file system connection descriptor in a metalanguage format comprising a local data system, a host system data structure, and a mapping data structure each comprising at least one tag. Moreover,

nowhere does the cited col. 19 teach or suggest tags in a metalanguage format that represent the local file system, the host file system; and a mapping between the local file system and the host file system. Instead, the cited Stedman includes a set of instructions in a page that is substituted for actual data, not encoding information in tags comprising a descriptor on a local system data structure, host system data structure and mapping as claimed.

The Examiner cited col. 7, lines 3-20 and col. 24, lines 8-44 of Imai and col. 2, lines 55-63 of Stedman as teaching the details concerning the file system connection descriptor. (Final Office Action, pgs. 3-4) Applicants submit that these cited sections of Imai and Stedman do not teach or suggest the claim requirements of encoding information in a metalanguage format on a file system connection descriptor comprised of tags representing a local file system, a host file system, and a mapping between a file in the local file system and a file in the host file system and a transfer type that defines a data format for transferring data between the host system and the local system to support remote editing of files in the host file system from the local file system

The cited col. 7 of Imai mentions that a file requesting client includes a connection unit for setting up a connection with a file server and a file requesting unit to request the file. A file receiving unit receives the file and file storage unit stores the file. Nowhere does this cited col. 7 anywhere teach or suggest encoding information in a metalanguage format on a file system connection descriptor comprised of tags representing a local file system, a host file system, and a mapping data structure comprising at least one tag representing a mapping between files in the local file system and files in the host file system and a transfer type that defines a data format for transferring data between the host system and the local system for this mapping. Instead, the cited col. 7 discusses client units to connect with a server, request a file, receive the requested file, display the file, etc. Although the cited col. 7 discusses how a file requesting client connects with a file server, there is no teaching or suggestion of a file descriptor having tags with information on a local system data structure, a host system data structure, a mapping data structure, and a transfer type. Thus, the specific claimed file system connection descriptor is not taught or suggested to support remote editing of files in the host file system from the local file system as claimed.

The cited col. 24 of Imai mentions that a user program requests the transfer and display of file. A request handling unit handles the display request and the file request unit transmits the request. The file server program transfers the requested file to the requesting client. A multiple file transfer program may be used to transfer multiple files to the client. Although the cited col. 24 discusses operations to request and transfer one or more files, nowhere does the cited col. 24 anywhere teach or suggest the claim requirements of a file system connection descriptor having tags in a metalanguage format representing a local file system, a host file system, a mapping between files in the local file system and files in the host file system, and a transfer type that defines a data format for transferring data between the host system and the local system for this mapping to support remote editing of files in the host file system from the local file system. Instead, the cited col. 24 concerns a file request and transfer operations to transfer a requested file from a server to client.

The cited col. 2, lines 55-63 of Stedman mentions that the client is linked to the server over the Internet and the server is linked to the host over an SNA network. Nowhere in this cited col. 2 of Stedman is there any teaching or disclosure of the claim requirement of a file system connection descriptor having tags in a metalanguage format representing a local file system, a host file system, and a mapping and transfer type as claimed.

Further, with respect to claims 3 and 16, the Examiner cited col. 28 , lines 38-65 of Imai as teaching the added claim requirement of a transfer type data structure storing a transfer type that defines how data will be transferred between the host system and the local system for this mapping. (Final Office Action, pg. 6) Applicants traverse with respect to this limitation, which was added to claims 1, 2, and 15.

The cited col. 28 of Imai discusses a transfer condition for preventing a wasteful transfer of a file of a type which cannot be utilized at the file request client. If a type of the selected file satisfies the transfer condition, the multiple files are transferred. If the type of the selected file does not satisfy the transfer condition, then another file is selected.

Although the cited col. 28 discusses a condition indicating a file type to not transfer, the cited Imai does not teach or suggest the claim requirement of a transfer type data structure storing a transfer type that defines a data format for transferring data between the host system and the local system for. Indicating file types not to transfer as in Imai is different from the

claim requirement of a transfer type defining a data format of how data will be transferred between a host and local system for a mapping between file systems.

The Examiner cited col. 19, lines 21-27 of Harvey as teaching the claim requirement with respect tot he preamended claim requirement. (Final Office Action, pg. 5) Applicants traverse with respect to the amended mapping claim.

The cited col. 19 of Harvey discusses a file converter that provides a mapping between different file formats. Although the cited col. 19 discusses converting files from one format to another, nowhere is there any teaching or suggestion of a transfer type that defines a data format for transferring data between the host system and the local system. Instead, the cited col. 19 discusses converting files between different file formats. Nowhere does this cited col. 19 teach or suggest a transfer type that defines a data format for transferring data between a host and local system. Moreover, nowhere does the cited col. 19 discuss the conversion to support remote editing of files in the host file system from the local file system. Thus, even if one were to modify the other references with Harvey to provide file conversion, this proposed modification still does not teach or suggest the claim requirement of a transfer type that defines a data format for transferring data between a host and local system to support remote editing of files in the host file system from the local file system.

Applicants submit that claims 1 and 15 are patentable over the cited combination because the cited Stedman, Imai, and Harvey, alone or in combination, does not teach or suggest all the requirements of claims 1 and 15.

Claim 2 recites a data structure embodied in a computer-readable storage medium, said data structure representing information describing a file system connection between a local file system located on a local system and a host file system located on a host system, wherein said data structure comprises a file system connection descriptor, said file system connection descriptor comprising: a local system data structure comprising at least one tag representing the local file system; a host system data structure comprising at least one tag representing the host file system; and a mapping data structure comprising at least one tag representing a mapping between a file in the local file system and a file in the host file system used to map and a transfer type that defines a data format for transferring data between the host system and the local system

to support remote editing of files in the host file system from the local file system, wherein the tags are encoded in a metalanguage format.

The Examiner cited col. 3, lines 38-49 of Stedman and col. 7, lines 4-23 and col. 8, lines 11-23 of Imai as teaching the requirements of claim 2. (Final Office Action, pg. 6) Applicants traverse.

The cited col. 3 of Stedman discusses multiple communication sessions having one browser application, one host extension object, one display control and one host computer, where a session ID is associated with each. The session ID is inserted in the hypertext link addresses. When the server receives a hypertext link from the client computer, the session ID is extracted and used to identify the host extension corresponding to the originating browser application.

Nowhere does the cited col. 3 anywhere teach or suggest information encoded in a metalanguage format on a file system connection descriptor comprised of tags representing a local file system, a host file system, and a mapping. Instead, the cited col. 3 concerns maintaining a session ID to use for multiple communication sessions. The particular claimed file system connection descriptor and tags are not disclosed in this cited col. 3.

The cited col. 7 of Imai mentions that a file requesting client includes a connection unit for setting up a connection with a file server and a file requesting unit to request the file. A file receiving unit receives the file and file storage unit stores the file. Nowhere does this cited col. 7 anywhere teach or suggest encoding information in a metalanguage format on a file system connection descriptor comprised of tags representing a local file system, a host file system, and a mapping between a file in the local file system and a file in the host file system used to map and a transfer type that defines a data format for transferring data between the host system and the local system to support remote editing of files in the host file system from the local file system, Instead, the cited col. 7 discusses client units to connect with a server, request a file, receive the requested file, display the file, etc. The specific claimed file system connection descriptor is not taught or suggested.

The cited col. 8 of Imai mentions a request handling unit to handle a file request and that the request processing unit searches a file list for the requested file corresponding to the file request from the file lists and transmits the searched file list to the file requesting client.

Although the cited col. 8 of Imai discusses how a request handling unit searches for a file, nowhere does the cited col. 8 anywhere teach or suggest tags in a metalanguage format representing a local file system, a host file system, and a mapping between the local file system and the host file system as claimed. There is no mention in the cited cols. 7 and 8 of Imai of a mapping between different file systems. Instead, the cited col. 8 discusses how a request for a file is handled.

Moreover, claim 2 additionally requires the claim requirement of mapping between a file in the local file system and a file in the host file system and a transfer type that defines a data format for transferring data between the host system and the local system to support remote editing of files in the host file system from the local file system. Applicants submit that this requirement provides additional grounds of patentability over the cited art for the reasons discussed above with respect to claims 1 and 15.

Applicants submit that claim 2 is patentable over the cited combination of art because the cited Stedman, Imai, and Harvey, alone or in combination, do not teach or suggest all the requirements of claim 2.

Claims 3-8, 9-14, and 16-21 are patentable over the cited art because they depend from one of claims 1, 2, and 15, respectively, which are patentable over the cited art for the reasons discussed above. The following dependent claims provide additional grounds of patentability over the cited art.

Amended claims 3, 9, and 16 depend from claims 2, 1, and 15, respectively, and further require that the mapping data structure comprises: a local file extension data structure storing a local file extension; and a host file pattern data structure storing a pattern describing a host file to which the local file extension will be applied.

Applicants amended these claims to remove the transfer type limitation, which was added to the independent claims 1, 2, and 15.

The Examiner cited col. 22, lines 16-53 of Imai as teaching the claim requirement of a host file pattern data structure storing a pattern describing a host file to which the local file extension will be applied. (Final Office Action, pg. 6) Applicants traverse.

The cited col. 22 mentions that a user requests the transfer and the display of a file by issuing a request and uses the URL to identify the file. A request handling unit in the client

handles the display, the file request unit transmits the file request, and the file server program transfers the file to the file requesting client. The client may receive a file list. The list includes files having identifiers requested by the user.

The cited col. 22 of Imai discusses handling of a file request between server and client. Nowhere does the cited col. 22 anywhere teach or suggest the claim requirement of a host file pattern data structure storing a pattern describing a host file to which the local file extension will be applied. Instead, the cited col. 22 discusses how a file or list of files matching a request is returned to the client.

Accordingly, claims 3, 9, and 16 provide additional grounds of patentability over the cited art.

Claims 4 and 17 depend from claims 1 and 15 and further require that the mapping data structure further comprises a host codepage data structure storing an identification of a host codepage in which data in the host file is encoded; and a local-codepage data structure storing an identification of a local codepage in which data in a local file is encoded.

The Examiner cited col. 27, lines 23-53 of Imai as teaching the additional requirements of these claims. (Final Office Action, pgs. 6-7) Applicants traverse.

The cited col. 27 mentions transferring only those files selected according to the file type to prevent waste due to the transfer of files that cannot be utilized at the file requesting client. In a third example, the multiple files transfer request unit is for transferring only those file which match the transfer condition provided in the file requesting client. Nowhere does this cited col. 27 anywhere teach or suggest a mapping data structure further including a host codepage data structure storing an identification of a host codepage in which data in the host file is encoded and a local-codepage data structure storing an identification of a local codepage in which data in a local file is encoded. Nowhere is there any mention or suggestion of host and local codepages as claimed.

Accordingly, claims 4 and 17 provide additional grounds of patentability over the cited art.

3. Added Claims 22-30

Added claims 22, 25, and 28 depend from claims 1, 2, and 15 and additional require a first transfer type indicates to transfer one file unmodified between the host file system and the local file system and wherein a second transfer type indicates to translate text in the file to transfer from the host file system to the local file system.

Added claims 23, 26, and 29 depend from claims 22, 25, and 28 and additionally require that the first transfer type comprises a binary transfer type and wherein the second transfer type comprises a text transfer type.

Added claims 24, 27, and 30 depend from claims 22, 25, and 28 and additionally require that a host and local code pages are used to translate text for the text transfer type.

The additional requirements of these added claims are disclosed on at least pg. 18, lines 23-27, which states the "storage element default-transfer 1226 stores the type of transfer of data between the remote host and the local workstation. A binary transfer transmits data unmodified between host and workstation; whereas, a text transfer translates translatable text using host and local code pages from one to another."

Applicants submit that the claims 22-30 are patentable over the cited art because they depend from one of claims 1, 2, and 15, which are patentable over the cited art for the reasons discussed above and because the additional requirements of these claims in combination with the base claims provide further grounds of patentability over the cited art.

Conclusion

For all the above reasons, Applicant submits that the pending claims 1-30 are patentable over the art of record and in compliance with the definiteness and enablement requirements. Applicants submit herewith a petition for a one month extension of time and the added claims. Nonetheless, should any additional fees be required, please charge Deposit Account No. 09-0460.

\\

\\

\\

\\

\\

The attorney of record invites the Examiner to contact him at (310) 553-7977 if the Examiner believes such contact would advance the prosecution of the case.

Dated: July 24, 2006

By: /David Victor/

David W. Victor
Registration No. 39,867

Please direct all correspondences to:

David Victor
Konrad Raynes & Victor, LLP
315 South Beverly Drive, Ste. 210
Beverly Hills, CA 90212
Tel: 310-553-7977
Fax: 310-556-7984